Putting MOSA Into Practice

A Modular Open Systems Approach (MOSA) is both a business and technical strategy for developing a new system or modernizing an existing one. As a business strategy, it enables program teams to build, upgrade and support systems more quickly and affordably. These benefits may be realized by leveraging the commercial sector investment in new technology through the use of corresponding commercial products available from multiple sources. As a technical strategy, MOSA is focused on a system design that is modular, has well defined interfaces, is designed for change and, to the extent possible, utilizes widely supported industry standards for key interfaces. As with any other approach, MOSA implementation should be based on upfront planning. To be most effective, the preparations for applying MOSA must be initiated early in the program and acquisition planning. Figure 3 depicts the essential elements and the supportive technical and business practices needed for effective MOSA implementation.

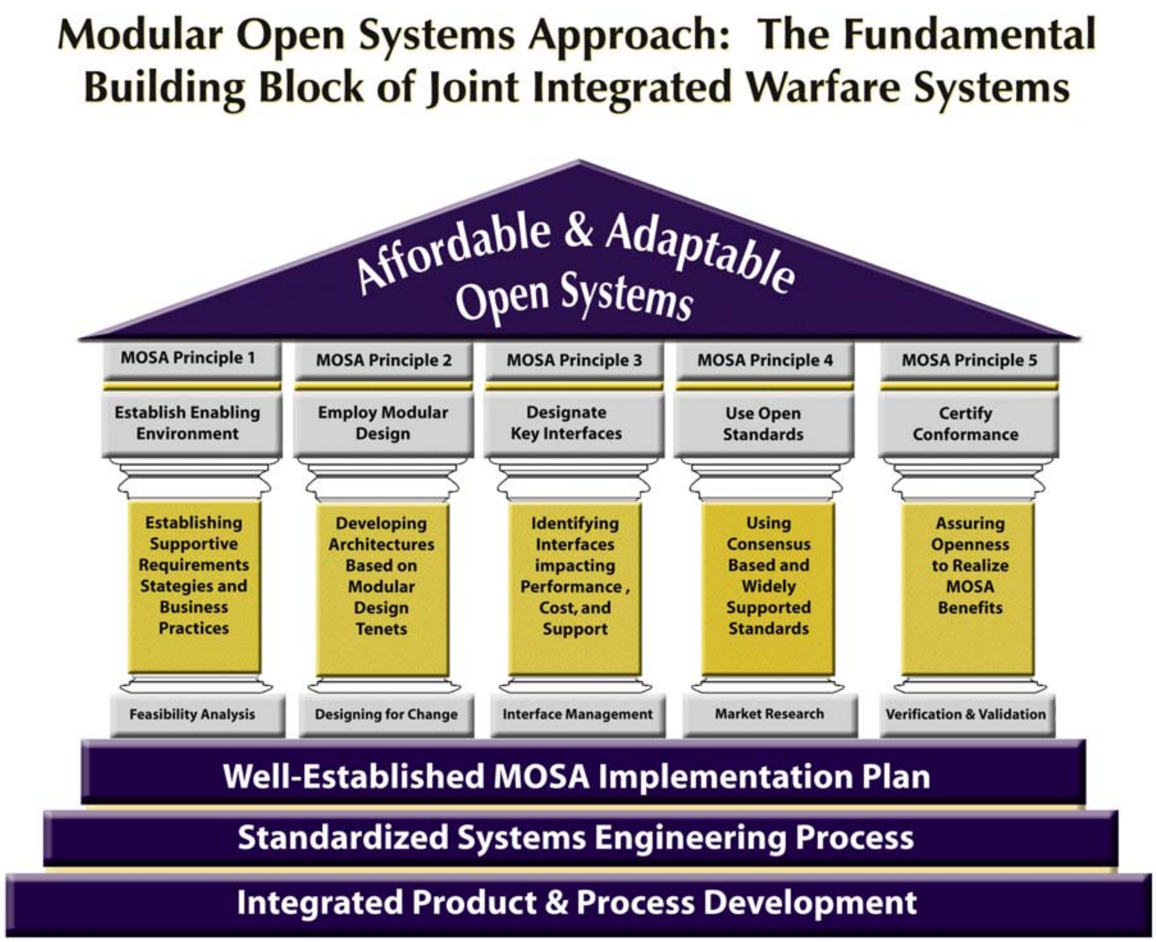

Figure 3: The MOSA Framework

As shown in Figure 3, effective implementation of MOSA requires employment of an Integrated Product and Process Development (IPPD) team approach, application of a sound systems engineering process, establishment of a MOSA implementation plan, and capitalizing on five proven MOSA principles: establishing enabling environment, employing modular design, designating key interfaces, selecting open standards, and certifying conformance to such standards.

The preferred strategy for applying MOSA is to employ an IPPD team comprised of government and industry representatives. The IPPD team should include all of the stakeholders involved in the acquisition, deployment, and employment of the system. The actual make up of an IPPD team is the responsibility of the program manager. At a minimum, an IPPD team should include those who require, specify, design, build, test, operate and maintain the system. The responsibilities of the IPPD team include establishing a tailored approach for implementing MOSA, identifying acquisition objectives and operational capabilities that lend themselves to the use of open systems, gathering and analyzing previous lessons learned, analyzing market research findings and other responsibilities relating to MOSA implementation.

The effectiveness of MOSA is also largely determined by the degree to which it is an integral part of a sound systems engineering (SE) process. We recommend incorporating MOSA into the SE process because it is during this process that MOSA has the greatest impact on the systems design and therefore the greatest benefit to the users of resulting products. Programs and contractors are encouraged to use SE standards such as EIA 632: Processes for Engineering a System, ISO 15288: Systems Engineering – System Life Cycle Processes or IEEE 1220: Standard for Application and Management of the Systems Engineering Process9 as the foundation for applying MOSA.

The MOSA implementation plan is a roadmap with specific objectives, tasks, principles, and milestones for putting MOSA into practice. The MOSA implementation plan should describe (1) how MOSA fits into a program's overall acquisition process and strategies for technology development, acquisition, test and evaluation, and product support; (2) what steps a program will take to analyze, develop, and implement a system or a system-of-systems architecture based on MOSA principles; and (3) how the program intends to monitor and assess its MOSA implementation progress.

The MOSA implementation plan should, at a minimum address the following five major tasks:

I. Identify and analyze capabilities and strategies that could most effectively be pursued by open system design solutions.
II. Assess the feasibility of open systems design solutions
III. Establish performance measures to assess MOSA implementation progress
IV. Use MOSA principles to develop an open architecture
V. Identify and resolve MOSA implementation issues and report the unresolved issues to Milestone Decision Authority.

For the remainder of this Guide, we will describe each of these tasks in greater detail.

I. Identify and Analyze MOSA Enabled Capabilities and Strategies

Program managers should identify specific operational and performance capabilities, and strategies for technology development, acquisition, test and evaluation, and product support that could be most effectively be enabled by using open systems design strategies. For example, the Program Management Office (PMO) could capitalize on open systems characteristics such as standardized interfaces and modular architectures to enhance joint combat capabilities, and support such evolving capabilities over their total life-cycle. The PMO could also select an open system solution if the warfighters require the ability to field innovative technologies as they become available to meet emerging threats and incorporate warfighter lessons learned in a timely fashion. Modular and scalable architectures could also be the preferred design choice if capabilities must be fielded in time-phased increments, and when the warfighter demands quick integration, interoperability, and reconfiguration of systems and their modules.

There are many acquisition strategies, operational capabilities, and performance requirements that lend themselves to the use of open systems in a program. The following list is a sample:

- Evolutionary acquisition and spiral development
- Operational requirements specified in an incremental manner over time.
- Requirements that place great emphasis on long-term sustainment and affordability, or establish affordability as the basis for fostering greater program stability.
- The ability to constitute and reconfigure functionally compatible forces and systems.
- Digitized battlefield or heavy reliance on digitized battlefield conditions to create operational capabilities.
- Receiving and disseminating command and control data in real time
- Seamless, high speed, digital information exchange among diverse warfighting elements.
- Overarching capabilities for a mission area that form a system of systems or family of systems.
- Reprogramming of software modules and communication networks where software reuse and increased flexibility is required.
- Integrated and modular communications and navigation capability.
- Application of an integrated approach for adding and facilitating the incorporation of future capabilities and advanced technologies with minimum impact on existing systems.
- Requirements that are defined in terms that enable and encourage offerors to supply commercial and non-developmental item equipment and call for minimizing the risks associated with being captive to specific products or sources.
- Future growth capabilities and performance characteristics that will be highly dependent on continuous use of emerging technologies such as computer, communication, surveillance, and navigation technologies.
- Interoperable joint service solutions and development of integrated architectures that must comply with open standards across different platforms.

- Modular contracting strategies

II. Assess the Feasibility of Open Systems Design Solutions

Open systems solution should not be pursued blindly. The PMO should make a business case for using open systems solutions. They should use a dynamic business case analysis model and apply market research findings to evaluate the appropriateness and feasibility of open systems. The model should take into consideration the changes in technology and threats to evaluate the total life cycle costs of designing the system as an open rather than a closed system. The PMO should conduct market research to identify technologies, standards, and compliant products needed for fulfilling capabilities and strategies identified at the preceding step. The review of technologies and standards will assist the PMO in the identification and assessment of risk areas with substantial impact on development, operation, and sustainment of a system over its life cycle. The PMO shall report to the MDA, in the context of the program Acquisition Strategy, the findings of the business case analysis used to justify non-compliance with the DoD MOSA policies and the potential economic impact on total ownership cost and risk to technology maturation and insertion over the service life of the system.

III. Establish Metrics to Assess MOSA Implementation Progress

In order to arrive at a system that exhibits open system characteristics, it is critical to have a set of measures or attributes indicating that these characteristics will be present as the system is being developed and when the system is complete. Establishing MOSA specific performance measures or at a minimum incorporation of MOSA principles in the program's performance measures is essential for realization of MOSA benefits. Program managers should use specific performance measures to gauge the progress on implementing MOSA, and ensure timely, efficient, and effective MOSA implementation. For example, the percentage of key interfaces defined by open standards could be used as a metric to measure the degree of system openness. Another example will be the percentage of obsolete modules in a system as a metric to measure the degree of system obsolescence. Other examples are the number or percentage of modules that can change without major system redesign; the percentage or actual total life cycle cost savings/avoidance attributable to compliance with MOSA principles; and number of latest technologies successfully migrated to a program as a result of adherence to MOSA principles.

IV. Use MOSA Principles to Develop an Open Architecture

Effective MOSA implementation is contingent upon proactive adherence to five major MOSA principles (please see Figure 1). These principles are essential parts of MOSA implementation plans and are fundamental to the design and implementation of open architectures. These guiding principles of MOSA are based on the experiences of programs that have implemented MOSA. Together, they are considered to be the minimum set of best business practices required for an effective application of MOSA. The first principle deals with creating an enabling environment characterized by supportive requirements, strategies, and business practices. The second principle is concerned with using modular design tenets to develop a robust architecture. The third principle focuses on designation of key interfaces so that we can distinguish among interfaces that are between technologically stable and volatile modules, between highly reliable and more frequently failing modules, and between modules with least interoperability impact and those that pass vital interoperability information. The fourth principle addresses the use of open standards for key interfaces, and the last principle pertains to certifying conformance to assure realization of open systems benefits.

For the remainder of this section, we will focus on describing these five MOSA principles. We will also address the MOSA indicators and briefly describe the tasks performed to effectively apply them in the overall system acquisition and SE process activities.

Principle 1: Establish an Enabling Environment

The first MOSA principle lays the foundation for establishing supportive requirements, business practices, and technology development, acquisition, test and evaluation, and product support strategies needed for effective development of open systems. Following is a list of type of supportive practices needed:

a. Existence of program requirements and system level functional and performance specifications that permit open systems development and will not impose design-specific solutions.
b. Presence of configuration management processes that encompass changes to key interfaces and corresponding standards over the system life cycle.
c. Existence of program staff with training or relevant experience in MOSA concepts and implementation.
d. Assignment of responsibilities for implementing MOSA. The PM should lead the efforts for MOSA implementation and should vigorously enforce the DoD MOSA policies and ensure that the PMO and contractors are aware of those policies and responsibilities for implementing them, understand MOSA concepts, and are familiar with the program MOSA implementation plan.
e. Program management and acquisition planning efforts conducive to MOSA implementation. The program's Systems Engineering Plan and strategies for technology development, acquisition, test and evaluations, and product support should all be supportive of MOSA implementation. For example, the Technology Development Strategy should emphasize the application of an open architecture so that the planned technology spirals or increments will be more effectively and affordably integrated. It should also emphasize that the prototype units that will be produced and deployed during technology development capitalize on open architecture attributes and benefits. The program's Systems Engineering Plan should encourage adherence to MOSA principles to ensure that the system is functionally partitioned into discrete, scalable, and reusable modules consisting of

isolated, self-contained functionally cohesive, interchangeable, and adaptable elements.

f. Effective identification and mitigation of barriers or obstacles that can potentially slow down or even, in some cases, undermine effective MOSA implementation.
g. Application of state of the art and widely used standard reference models and architectural frameworks adopted by the industry.
h. Continuing market research and analysis. Programs should establish a process for continuous market research to:

    - Analyze commercial market capabilities and future market and technology trends.
    - Collect and evaluate data on available and emerging interface standards to determine whether or not they are applicable to their particular system.
    - Assess the breadth of open and de facto standard compliant products to determine if suppliers will continue to produce or support the standards selected.
    - Help in partitioning the system into modules based on widely-supported commercial interfaces.

It should be noted that since there is no single best approach for implementing MOSA; therefore, every program should establish its own approach tailored to their specific acquisition strategy and program requirements.

Principle 2: Employ Modular Design

Modular design provides for expansion or functional reconfiguration through the incorporation of replaceable modules. A simple example is that of a desktop computer. The functions may include word processor or graphics applications depending on the software modules installed. Similarly, other functionally may be easily added by installing a new hardware module such as a modem. These functions are possible because sufficient aspects of the corresponding system interfaces are well defined so that designers of individual modules can do their work independently. Under the ideal situation, multiple product choices are available that can be inserted with minimal integration.

The first step in design of a new system is to partition the system into functions and identify the functional elements that should be modularized. The process then proceeds to decomposing higher-level functions into lower-level functions, identifying interfaces (e.g., internal and external), and finally to allocating performance from higher to lower-level functions. This process is repeated to define successively lower level functional and performance requirements, thus defining modular architectures at ever-increasing levels of detail.

For a legacy system, the focus will be on gathering information on the existing or as-built design, and performing the essential modular partitioning and mapping of services and interfaces to known functions and capabilities. To assess the appropriateness of a modular standards-based architecture, several items such as design specifications, interface control documents, functional specifications, and known standards profiles for an existing system may be reviewed. Knowledge of the other respective systems/subsystems that must be interfaced is derived from the existing requirement documents.

IPPDs must use modularity principles (maximal cohesiveness of the functions and minimal coupling among elements) to convert functional architectures to design architectures. They need to group and regroup components that perform a single independent function or single logical task into modules. They must also use desirable attributes such as low coupling, high binding (cohesion), and low connectivity to do the grouping required for modularity. Decoupling modules eases development risks and makes future modifications easier. High binding (similarity of tasks performed within the modules) allows for use of identical or like components or for use of a single component to perform multiple functions. Low connectivity (relationship among internal elements of one module to those of another module) is desirable because it reduces design and test complexity.

IPPDs should also prototype the system, subsystems, and components to demonstrate the integration of the system using the proposed modular decomposition. They should also use prototypes to demonstrate standards and standards-compliant products. Final products should not be selected at this time and the IPPD should demonstrate that potential interface standards and specifications will achieve required system performance.

Principle 3: Designate Key Interfaces

A key interface is an interface for which the preferred implementation uses an open standard to design the system for affordable change, ease of integration, interoperability, commonality, reuse or other essential considerations such as criticality of function. The IPPD should evaluate the system modules using the characteristics listed above to designate an interface as a key interface. The IPPD should recursively apply the evaluation characteristics to the top-level design components/modules and their sub-modules until all key interfaces are designated.

Programs must determine the level of implementation (e.g., subsystem, system, system-of-systems) at and above which they aspire to maintain control over the key interfaces and would like these interfaces to be defined by widely supported and consensus based standards. To establish the desired level of control, programs must review the results of market analysis on the availability of open standards for selected key interfaces and assess the impact of a chosen level of control on long-term viability and affordability of the system. Defining the level of interface control too low may limit efficient technology insertion, while defining the level too high may lead to the use of proprietary interfaces for major system components resulting in limited supplier support. Programs should use a business case analysis to justify the use of open standards for such interfaces at desired levels of implementation.

Programs may use Work Breakdown Structures (WBS) developed from the design architecture or a reference model to designate key interfaces. Reference models are perhaps the best means for designating key interfaces. IPPDs should first determine whether there is an existing reference model they can use, or if they need to develop one appropriate to the concept(s) under consideration. A technical reference model (TRM) provides a high level, generalized system view of the weapon system family. Generally speaking, a TRM:

- Is a common high-level communications vehicle for system stakeholders. It embodies the earliest set of design decisions about the system. These decisions are the most difficult to get right, the hardest ones to change and have the most far-reaching effects downstream.
- Forms the organizational plan for development of a modular, open system. It establishes a context for understanding how disparate technologies and standards relate to each other. Done well, a reference model is a high-level vehicle for incorporating existing or planned components.

- Provides a framework for breaking out the system and applying standards. Well-formed reference models exhibit modularity. The reference model provides a framework for how to apply standards, particularly, how to identify interfaces that are key to achieving system technical and business goals.

Reference models provide a high-level view of the system modularity and the interfaces between those modules. Figure 4 is an example of how a reference model might depict the functional parts comprising systems belonging to an aircraft. It demonstrates decomposition of the overall weapon system's mission into a smaller number of simpler functional building blocks. Each functional building block can be similarly decomposed. The selection of particular functional entities represents the initial design decisions for how the weapon system will be engineered. Here, modularity in design is facilitated by aligning functional partitioning with physical modularity where modularity is used to facilitate the replacement of specific subsystems and components without impacting other parts of the system. The boundary or interface between each building block pair is defined by the services provided over that interface.

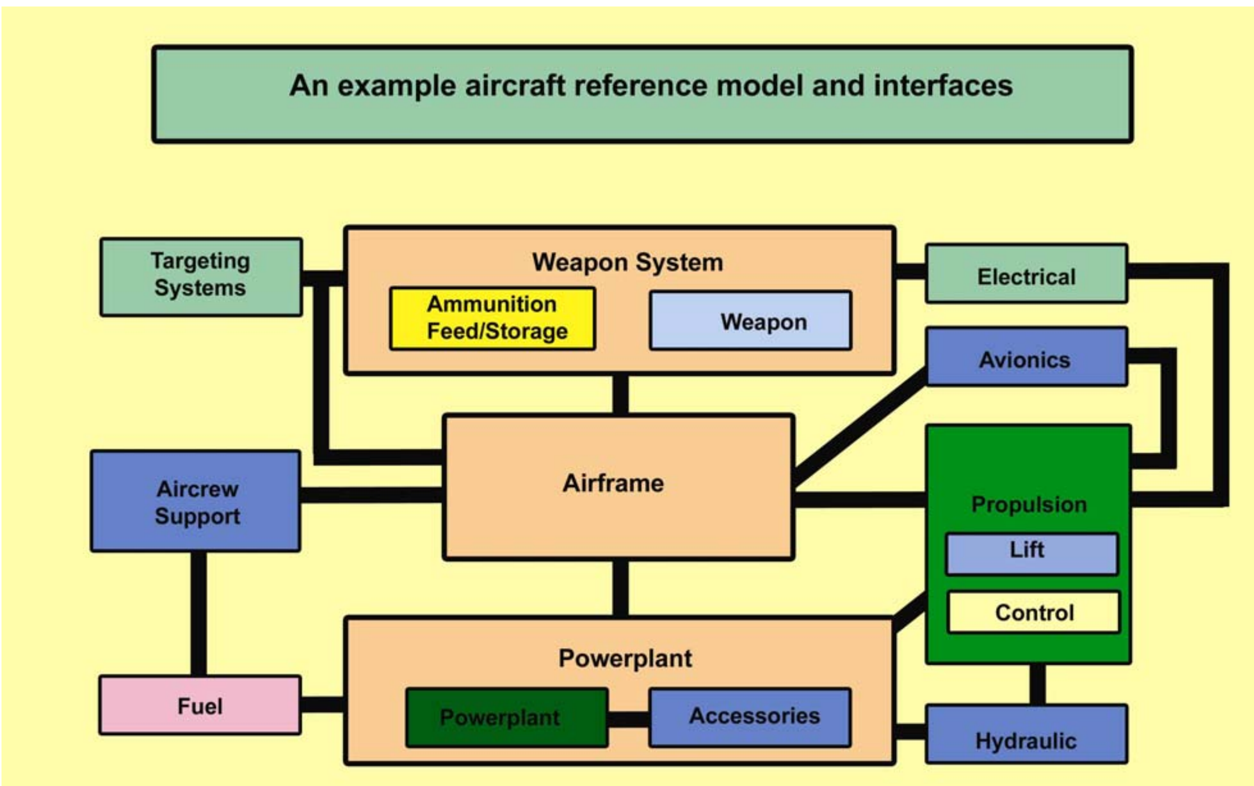

Figure 4: An Aircraft Reference Model

Principle 4: Use Open Standards

Once key interfaces are identified, the next task for the program team is to determine whether or not it is feasible to use an open interface standard for each of the key interfaces. In MOSA, the fact that an interface has been designated as a key interface means that the preferred implementation would employ an open interface standard. This does not mean that the final implementation for every key interface will always use an open standard. There will be times when the best decision is to use a proprietary standard. This decision is left to the program manager. The following factors may be considered by the IPPD in using open standards for key interfaces:

- Overall acquisition strategy (e.g., the likelihood that the technologies/engineering for full capability still need to be developed and whether or not the longer-term requirements are stable or addressed as evolving increments.)
- The degree of dependency on rapidly evolving technology and the technology readiness level for the components or items at both ends of an interface
- The intensity and magnitude of risks associated with a proprietary interface standard
- Need for minimizing integration risks over the life of the system
- Need to take advantage of competition throughout the life cycle
- Need for design flexibility, modularity, and interface control
- Availability, maturity, verification, and accreditation of standards for an interface
- Support strategy (e.g., the extent of market acceptance and availability of products that comply with a selected standard)

Key interfaces should be examined very carefully to insure that the use of an open standard is both feasible and appropriate based on performance and business objectives. As previously mentioned, the aim of MOSA is not to make all the internal and external system interfaces open. There are significant advantages to using open interface standards; however they should only be used if it makes sense within the context of the performance and business objectives of the particular program. The utilization environment of a system also has some open system implications. The physical environment may also necessitate modification of commercial products because they may not withstand the humidity, temperature, vibration, and electromagnetic environments in a weapon system. Long term supportability and maintainability may also be impacted if unique proprietary interface standards are employed in the system resulting in dependency on a sole source and possibly costly maintenance for the life of a system. The size and weight of products in the system under development may impose restrictions on use of commercial interface standards and products. But, the benefits gained from the isolation of the function for future change may far outweigh any disadvantage of using a proprietary interface standard due to utilization environment constraints. If the use of an open standard is not appropriate at a given time, the program should continue to look for future opportunities within the system to take advantage of benefits of using open standards. As a general rule, system developers should only use open standards when it makes business and technical sense.

Interfaces may be controlled through interface management. Interface management identifies, develops, and maintains the external and internal interfaces necessary for system operations. It also ensures that system elements are compatible in form, fit, and function. Interface management may also include establishing an Interface Control Working Group, which among other things may establish the Interface Control Documentation.

Once standards are selected for the appropriate key interfaces, the IPPD also needs to develop a method of verification or conformance to the interface specification itself. IPPDs need to verify claims made by vendors that their products comply with certain interface standards and their respective profiles. Test suites should be developed to ensure conformance of selected commercial items and non-developmental items to appropriate interface definitions.

Principle 5: Certify Conformance

The program manager, in coordination with the user, should prepare validation and verification mechanisms such as conformance certification and test plans to ensure that the system and its component modules conform to the external and internal open interfaces allowing plug-and-play of modules, net-centric information exchange, and re-configuration of mission capability in response to new threats and technologies. Such plans must become an integral part of the overall organization change management process. As systems evolve through spiral development and in response to changes in requirements and technologies, external and internal interfaces will most likely change which necessitates proactive conformance and integration tests. The conformance test and certification plans should ensure that the system components and selected commercial products avoid utilization of vendor-unique extensions to interface standards and can easily be substituted with similar components from competitive sources.

V. Identify and Resolve MOSA Implementation Issues and Report the Unresolved Issues to Milestone Decision Authority

MOSA is characterized by an enabling environment, employment of modular design, designation of key interfaces, use of open interface standards and certification of conformance. In order to arrive at a system that exhibits these open system characteristics, it is critical to establish a procedure to assess MOSA implementation progress, identify the implementation issues, and satisfactorily resolve such issues. The procedure should be based on a set of measures or attributes indicating that the characteristics associated with each MOSA principle will be present as the system is being developed and when the system is complete. These measures or attributes are intended for consideration and use by acquisition executives, program managers and IPPD teams responsible for implementing MOSA to assure the achievement of MOSA benefits. The OSJTF has developed a set of indicators, in the form of MOSA implementation questions (please refer to Appendix C), to help assess the extent to which MOSA is implemented in an acquisition program, and also to identify actual or potential MOSA implementation issues. These questions are representative of the actual questions used in the MOSA Program Assessment and Review Tool (PART), which is an automated analytical tool that relies on objective, data evidence-based judgments to assess and evaluate MOSA implementation. The MOSA PART is an adaptation of the OMB Program Assessment Rating Tool (PART), which is a questionnaire designed to provide a consistent approach to rating programs across the Federal government. The responses to the questions, provided on the MOSA Implementation Questions tab of the PART, will be evaluated to determine the overall implementation level of MOSA, identify actual and potential implementation issues, and determine individual areas where improvements might be made. The evaluation results are shown in the Assessment Report tab of the PART. Program managers can use either MOSA PART or other tools to identify specific MOSA implementation issues that their Integrated Product Team must address and satisfactorily resolve. In case such issues can not be resolved at the lower level, program managers must report them to the Milestone Decision Authority for final resolution.

The MOSA PART can be found on the Open Systems Joint Task Force website at http://ww.acq.osd.mil/osjtf.

Programs should proactively identify the problems, obstacles and issues they encounter in implementing their MOSA. They can use a variety of means (e.g., the MOSA PART questions; market research findings; doctrine, organization, training, materiel, leadership and education, personnel and facilities (DOTMLPF) implication assessments; MOSA specific metrics; etc.) to discover potential problems, obstacles, or barriers toward MOSA implementation. For example, the marketing research can point to the fact that an emerging open standard for a key interface may not be matured by the time predicted, or the test suits for its conformance testing may not be available when needed. Under this circumstance, the program may not have any other choice but to use a de-facto or proprietary standard for that interface to resolve the issue. The programs should undertake the following procedure to effectively deal with a major MOSA implementation issue (problem, obstacle, or barrier):

1. Use appropriate means to discover MOSA implementation problems, obstacles, and barriers.
2. Analyze and document the problems, obstacles, or barriers, and the circumstances surrounding them.
3. Address the identified problems, obstacles, or barriers via the Integrated Product Team (IPT) process and focus on resolving them at the lower IPT levels. For example, propose a migration plan to attain future openness.
4. Present the identified problems, obstacles, or barriers as issues to the MDA only when unresolved at a lower level. The issues must be reported to the MDA in the context of a summary within the program acquisition strategy. The summary should describe the potential economic impact of the issue (e.g., using closed interfaces) on total ownership costs and the risk to technology maturation and insertion over the service life of the system. The summary should also describe the potential impacts of the issue on the ability to integrate and/or retrofit earlier increments with later increments, and the effects on integrating the system with other systems within a system of systems context.